

## ЛАБОРАТОРНА РОБОТА № 6

### Паралельне програмування

**Мета:** засвоїти принцип проектування програми з використанням паралельного програмування; ви-вчити особливості застосування: go-рутин, каналів, синхронізації go-рутин

#### Хід роботи:

**Завдання 1:** Оголосив структуру Bank, яка представляє інформацію про наявність грошей, які зберігаються у банку. Має наступні поля

```
type Bank struct {
    name      string
    bankMoney float64
    deposit   float64
    credit    float64
    clients   []*Client
}
```

Оголосив структуру Client, яка містить наступні поля

```
type Client struct {
    name      string
    surname   string
    accountNumber string
    cDeposit  float64
    cCredit   float64
    bank      *Bank
}
```

**Завдання 2:** Для кожної структури реалізував конструктори.

**Завдання 3:** Реалізував set- та get- методи для кожної структури

**Завдання 4:** Реалізував необхідні методи для структур.

**Завдання 5:** Виконав усі необхідні перевірки на можливість отримання грошей з рахунку

**Завдання 6:** Створив консольне меню з наступними пунктами:

```
Оберіть пункт меню:
1 - Створити банк
2 - Додати клієнта для кредитів
3 - Додати клієнта для депозитів
4 - Вивести інформацію про клієнта за прізвищем
5 - Вивести інформацію про клієнта за номером акаунту
6 - Вивести інформацію про банк та усіх клієнтів
Інший символ - Завершити роботу програми
Введіть обраний пункт меню:
```

					ДУ «Житомирська політехніка».22.121.27.000 – Лр6			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Шеренгівський О.О.			Звіт з лабораторної роботи		Лім.	Арк.
Перевір.		Петросян Р.В.						1
Керівник								5
Н. контр.							ФІКТ Гр. ІПЗк-22-1	
Зав. каф.								

## Результат виконання програми

```
PS E:\Політех\go\Lab6> go run main.go
warning: GOPATH set to GOROOT (C:\Users\PC\sdk\go1.19) has no effect
Lab6. Паралельне програмування
```

Оберіть пункт меню:

- 1 - Створити банк
- 2 - Додати клієнта для кредитів
- 3 - Додати клієнта для депозитів
- 4 - Вивести інформацію про клієнта за прізвищем
- 5 - Вивести інформацію про клієнта за номером акаунту
- 6 - Вивести інформацію про банк та усіх клієнтів

Інший символ - Завершити роботу програми

Введіть обраний пункт меню: 1

Введіть назву банку: Tohsaka

Введіть кількість грошей на рахунках банку: 999999

Введіть обраний пункт меню: 2

Введіть ім'я користувача: Артем

Введіть прізвище користувача: Гордєєв

Введіть номер акаунту користувача: 2001

Введіть ім'я користувача: Саша

Введіть прізвище користувача: Ковальчук

Введіть номер акаунту користувача: 578

Введіть обраний пункт меню: 6

Оберіть пункт меню:

- 1 - Створити банк

Банк: Tohsaka

Депозити: 600.00

Кредити: 0.00

Баланс: 997999.00

Клієнти

					ДУ «Житомирська політехніка».22.121.27.000 – Лр6	Арк.
		Левківський В. Л.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

```
CLIENT INFO
FullName: Гордєєв Артем
Deposit: 0.00
Credit:0.00
FullName: Ковальчук Саша
Deposit: 600.00
Credit:0.00
Account number:578
```

4

Введіть прізвище користувача: Ковальчук

```
CLIENT INFO
FullName: Ковальчук Саша
Deposit: 0.00
Credit:0.00
Account number:578
```

Введіть номер акаунту користувача: 2001

Оберіть пункт меню:

- 1 - Створити банк
- 2 - Додати клієнта для кредитів

```
CLIENT INFO
FullName: Гордєєв Артем
Deposit: 0.00
Credit:0.00
Account number:2001
```

**Висновки:** в ході виконання лабораторної роботи засвоїв принцип проектування програми з використанням паралельного програмування; ви-вчив особливості застосування: go-рутин, каналів, синхронізації go-рутин

					ДУ «Житомирська політехніка».22.121.27.000 – Лр6	Арк.
		Левківський В. Л.				
Змн.	Арк.	№ докум.	Підпис	Дата		3

## Контрольні питання

### 1. Що таке процес?

*Процесом називають поточну програму і всі її елементи: адресний простір, глобальні змінні, реєстри, стек, відкриті файли і так далі.*

### 2. Що таке потік?

*Потік виконання – найменша одиниця обробки. Потік виконання знаходиться всередині процесу. Кілька потоків виконання можуть існувати в рамках одного і того ж процесу і спільно використовувати ресурси, тоді як процеси не поділяють цих ресурсів, а взаємодія відбувається за допомогою спеціальних механізмів*

### 3. Які особливості потоків у мові Go?

*У Go паралельне програмування засноване на моделі CSP (Communicating Sequential Processes – взаємодіючі послідовні процеси). У мові Go є дві особливості. Перша: go-підпрограми – легковагі потоки виконання. Друга: канали, що забезпечують надійний засіб одно- і двостороннього обміну даними між go підпрограмами*

*Особливості go-підпрограм:*

- незалежно виконується функція, запущена за допомогою інструкції go;
- наявність власного стека;
- невеликі витрати по пам'яті ~ 2 Кб-4 Кб;
- Runtime Go займається розподілом горутин по процесам.

### 4. Як створити потік в мові Go?

*Горутини можна уявляти як «легковисні потоки», щоб створити горутину потрібно просто поставити ключове слово go перед кодом виклику функції. Щоб продемонструвати наскільки це просто, давайте створимо дві функції пошуку, викличемо їх з ключовим словом go і друкуватимемо повідомлення кожного разу, коли вони знайдуть «руду» у своїй копальні.*

*Канали дозволяють горутин обмінюватися даними. Це своєрідна труба, через яку горутини можуть посилати та приймати інформацію від інших горутин. Читання та запис у канал здійснюється за допомогою оператора-стрілочки (<-), який вказує на-прямок руху даних.*

### 5. Для чого призначені канали?

*Канали – це механізм взаємодії між go-підпрограмами, які паралельно виконуються. При використанні каналів синхронізація кінців каналу (і відповідних їм go-підпрограм), які відправляють і приймають, виконується в момент взаємодії*

					ДУ «Житомирська політехніка».22.121.27.000 – Лр6	Арк.
		Левківський В. Л.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

## 6. Призначення керуючої конструкції select?

Інструкція *select* використовується для вибору каналу, який виконав транзакцію. По-  
хожа на оператор *switch*. Має наступний вигляд:

```
select {  
  case канал1:  
    блок1  
  ...  
  case каналN:  
    блокN  
  default:  
    блокD  
}
```

					ДУ «Житомирська політехніка».22.121.27.000 – Лр6	Арк.
		Левківський В. Л.				5
Змн.	Арк.	№ докум.	Підпис	Дата		